

Applicant : Yuh-Cherng Wu, Ph.D., et al.
Serial No. : 10/782,576
Filed : February 19, 2004
Page : 5 of 11

Attorney's Docket No.: 13906-090001 / 2003P00410US

REMARKS

In the non-final office action mailed April 13, 2006, the Office Action rejected claims 1-15. Applicants have amended claim 15. As such, claims 1-15 remain pending. Applicants respectfully request the Examiner's reconsideration in view of the amendments and remarks below.

CLAIM REJECTION- 35 USC 101

The Office Action rejected claim 15 under 35 USC § 101 as being directed to non-statutory subject matter. In accordance with the Examiner's suggestion, Applicants have amended the preamble of independent claim 15 to clarify that the claimed computer program *product* is tangibly embodied in an information carrier. This amendment finds support at least in the specification at paragraph [0072]. Accordingly, Applicants respectfully request that the Examiner reconsider and remove the rejection of claim 15 under 35 USC § 101.

CLAIM REJECTIONS – 35 USC 103

The office action rejected claims 1-15 as obvious under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. 5,386,498 to Kakefuda ("the Kakefuda reference") in view of U.S. Publ. Pat. Appl. 2004/0122799 to Goyal et al. ("the Goyal reference"). Claims 1 and 15 are independent.

Applicants submit that independent claims 1 and 15 define subject matter that is patentable over the cited references because neither the Kakefuda nor the Goyal references teach or suggest any of (1) determining an application programming interface (API), (2) assessing combinations of candidate tables and candidate APIs, or (3) selecting a master API for a new knowledge base, all of which are required by Applicants' independent claims. In fact, nowhere does either cited reference mention or even contemplate APIs or application program interfaces.

Applicants' claims are directed to methods and apparatus that can increase the utility of knowledge bases by enabling non-expert users to quickly and easily create custom knowledge bases without the need for in-depth knowledge about the data sources used to create the new

Applicant : Yuh-Cherng Wu, Ph.D., et al.
Serial No. : 10/782,576
Filed : February 19, 2004
Page : 6 of 11

Attorney's Docket No.: 13906-090001 / 2003P00410US

knowledge base. In particular, Applicants' claimed system is capable of automatically generating a new custom knowledge base with an API that provides a software interface to enable a wide variety of application programs in an integrated system (e.g., an enterprise system) to retrieve, modify, and/or store data in the knowledge base. (Paras. [0014, 0025].) In general, an API supplies an application program with metadata (e.g., format, data type, etc.) needed to reach the contents of the knowledge base. (Para. [0003].) Unlike a skilled human expert/programmer, who might generate an API directly based upon detailed understanding of the data objects required for the new knowledge base, Applicants' disclosed system generates a new knowledge base and API by applying heuristic rules to identify pre-existing data and pre-existing APIs that are suitable for the new knowledge base. (FIG. 3, #110 – 114.) Specifically, a data exploring process explores pre-existing data to identify candidate database tables that are associated with (e.g., contain format, content, data type) data objects that may be relevant for the new knowledge base. The data exploring process applies heuristic rules to the candidate database tables. (Para. [0021-0022, 0031].) A code exploring process searches for pre-existing API code that may be re-used in the new knowledge base. (Para. [0022, 0033].) The code explorer process applies heuristic rules to evaluate combinations of candidate database tables and APIs to determine a suitable API that enables access by any suitable application program in an integrated system.

Applicants' claim 1 is directed to a method for generating a new knowledge base. The method includes receiving a signal that defines content to be included in the new knowledge base. The method also includes searching for tables within an existing database and identifying at least one candidate table that is associated with data that may be relevant to the defined content to be included in the new knowledge base. The method further includes determining at least one candidate application programming interface (API) for each candidate table, and assessing combinations of the identified at least one candidate table and the determined at least one candidate API and selecting therefrom a master table and a master API for the new knowledge base. The method further includes generating the new knowledge base that is

Applicant : Yuh-Cherng Wu, Ph.D., et al.
Serial No. : 10/782,576
Filed : February 19, 2004
Page : 7 of 11

Attorney's Docket No.: 13906-090001 / 2003P00410US

accessible by more than one application program within an integrated system. Claim 15 is directed to a computer program product with similar subject matter.

The cited references, taken alone or in combination, do not anticipate or render obvious any of Applicants' claims.

Instead of disclosing Applicants' claimed invention, the Kakefuda reference is directed to an expert system that can synthesize knowledge bases using an inference engine, a synthesis knowledge base synthesizing unit, and a certainty factor determination unit. (Col. 1, ln. 58 – col. 2, ln. 19.) The Office Action correctly states that the Kakefuda reference “does not teach:”

... determining at least one candidate application programming interface (API) for each candidate table; assessing combinations of the identified at least one candidate table and the determined at least one candidate table and the determined at least one candidate API and selecting therefrom a master table and a master API for the new knowledge base; and generating the new knowledge base that is accessible by more than one application program within an integrated system.” (Office Action at page 3.)

The Goyal reference does not overcome the deficiencies of the Kakefuda reference. Instead of disclosing the teachings that the Office Action acknowledges are missing from the Kakefuda reference, the Goyal reference is directed to a storage system with a policy manager. (Abstract.) Based on a policy in the policy manager, a database creates a tablespace for itself and establishes a storage allocation. (Para. [0013].) If a transaction causes the tablespace to run out of storage, the system taught by the Goyal reference automatically extends the tablespace to prevent the pending transaction from failing.

Neither of the cited references, when taken alone or together, teaches or suggests generating a new knowledge base in the manner recited in Applicants' claim 1 or claim 15. For example, neither reference suggests or even contemplates (i) determining an API, (ii) assessing combinations of at least one candidate table and at least one candidate API, or (iii) selecting a master API, as required by both independent claims. In fact, neither reference makes a single mention of an API. As such, the Office Action does not set forth a prima facie case of as to each element recited in Applicants' independent claims. For the following reasons, Applicants

Applicant : Yuh-Cherng Wu, Ph.D., et al.
Serial No. : 10/782,576
Filed : February 19, 2004
Page : 8 of 11

Attorney's Docket No.: 13906-090001 / 2003P004101JS

respectfully request that the Examiner reconsider and remove the obviousness rejections based on the cited references.

First, neither reference, when taken alone or in combination, teaches or suggests "determining at least one candidate application programming interface (API) for each candidate table," as required by Applicants' claims.

The Office Action contends that the policies in the policy manager to communicate with the database, as disclosed in the Goyal reference, are the same as Applicants' recited "determining at least one candidate application programming interface (API) for each candidate table." (Office Action at pgs. 3-4.) In addition, the Office Action equates the API in Applicants' claims with the policies in the policy manager as described in the Goyal reference.

Applicants disagree with both of these contentions. As to the latter contention, the policy described in the Goyal reference is not similar to an API. As pointed out above, an API supplies an application program with metadata (e.g., format, data type, etc.) needed to reach the contents of the knowledge base. In contrast, the policies in the policy manager of the Goyal reference result in creation of a certain size allocation in a tablespace. Determining a size allocation of tablespace does not teach, suggest, or even contemplate an API that provides metadata to enable application programs to access data in a knowledge base. As such, the latter contention fails.

Accordingly, Applicants submit that the former contention also fails because the policies in the policy manager to communicate with the database do not teach or suggest "determining at least one candidate application programming interface (API) for each candidate table," as required by Applicants' claims. An API and a size allocation policy are quite different. Without a proper API, for example, a policy to vary the size of an allocation table as taught by the Goyal reference would not enable an application program to read or write data in a new knowledge base generated in accordance with Applicants' claims. Moreover, the Office Action does not point to any teaching or suggestion in any prior art reference for performing the recited determining step for each candidate table.

Applicant : Yuh-Cherng Wu, Ph.D., et al.
Serial No. : 10/782,576
Filed : February 19, 2004
Page : 9 of 11

Attorney's Docket No.: 13906-090001 / 2003P00410US

Second, neither reference, when taken alone or in combination, teaches or suggests "assessing combinations of the identified at least one candidate table and the determined at least one candidate API," as required by Applicants' claims.

The Office Action does not set forth any disclosure in the cited references that teaches or suggests several elements of this element of Applicants' claims. For example, as described above, the Goyal reference does not teach or suggest "determining at least one candidate API;" therefore, the Goyal reference does not teach or suggest "assessing combinations of ... the determined at least one candidate API" as recited in Applicants' claims.

In addition, the Office Action does not point to any teaching or suggestion in the Goyal reference of "identifying at least one candidate table;" therefore, the Goyal reference does not teach or suggest "assessing combinations of ... the identified at least one candidate table" as recited in Applicants' claims.

Moreover, the Office Action does not point to any teaching or suggestion of "assessing combinations" as recited in Applicants' claims. The Office Action is silent as to how the Goyal reference discloses generating or assessing combinations, or what is being combined and assessed.

Third, neither reference, when taken alone or in combination, teaches or suggests "selecting therefrom a master table and a master API for the new knowledge base," as required by Applicants' claims.

As to this element of Applicants' claims, the Office Action contends that "determining tablespace based on policy rules," as disclosed in the Goyal reference, is the same as "determining a master table and a master API." (Office Action at pg. 4.)

Applicants disagree. As described above, tablespace or policy rules as described by the Goyal reference do not teach or suggest an API. Neither do tablespace or policy rules as described by the Goyal reference teach or suggest a master table. Applicants' claims require a master table selected from "at least one candidate table that is associated with data that may be relevant to the defined content to be included in the new knowledge base." The Goyal reference

Applicant : Yuh-Cherng Wu, Ph.D., et al.
Serial No. : 10/782,576
Filed : February 19, 2004
Page : 10 of 11

Attorney's Docket No.: 13906-090001 / 2003P00410US

does not, and the Office Action has not contended that it does, teach or suggest the master table required in Applicants' claims.

For each of the foregoing reasons, the Office Action has not set forth a prima facie case of obviousness as to either independent claim.

Applicants submit that independent claims 1 and 15 are patentable over the cited references. As such, dependent claims 2-14 are also patentable over the cited references. Applicants submit that claims 1-15 are in condition for allowance, and respectfully request that the Examiner reconsider and remove the rejections under 35 USC 103 as to each of these claims.

CONCLUSIONS

Applicants submit that claims 1-15 (as amended) are in condition for allowance, and respectfully request that the Examiner reconsider the pending claims, and promptly issue a notice of allowance of all claims 1-15.

It is believed that all of the pending claims have been addressed. However, the absence of a reply to a specific rejection, issue or comment does not signify agreement with or concession of that rejection, issue or comment. In addition, because the arguments made above may not be exhaustive, there may be reasons for patentability of any or all pending claims (or other claims) that have not been expressed. Finally, nothing in this paper should be construed as an intent to concede any issue with regard to any claim, except as specifically stated in this paper, and the amendment of any claim does not necessarily signify concession of unpatentability of the claim prior to its amendment.

Applicant : Yuh-Cherng Wu, Ph.D., et al.
Serial No. : 10/782,576
Filed : February 19, 2004
Page : 11 of 11

Attorney's Docket No.: 13906-090001 / 2003P00410US

This response is accompanied by a Petition for One-Month Extension of Time. Please apply any other charges or credits to deposit account 06-1050.

Respectfully submitted,

Date: 3 Aug. 2006

Craig O. Thompson
Craig O. Thompson
Reg. No. 47,990

Fish & Richardson P.C., P.A.
60 South Sixth Street
Suite 3300
Minneapolis, MN 55402
Telephone: (612) 335-5070
Facsimile: (612) 288-9696

60356382